

---

# Contents

## Part I Parameterized Tractability

<b>1 Preliminaries</b> . . . . .	3
1.1 Basic Classical Complexity . . . . .	3
1.2 Advice Classes . . . . .	7
1.3 Valiant–Vazirani and BPP . . . . .	9
1.4 Historical Notes . . . . .	12
1.5 Summary . . . . .	13
<b>2 The Basic Definitions</b> . . . . .	15
2.1 The Basic Definition . . . . .	15
2.2 The Other Flavors of Parameterized Tractability . . . . .	16
2.3 Exercises . . . . .	19
2.4 Historical Notes . . . . .	20
2.5 Summary . . . . .	21

## Part II Elementary Positive Techniques

<b>3 Bounded Search Trees</b> . . . . .	25
3.1 The Basic Method . . . . .	25
3.2 VERTEX COVER . . . . .	25
3.2.1 A Basic Search Tree . . . . .	25
3.2.2 Shrinking the Search Tree . . . . .	26
3.3 PLANAR INDEPENDENT SET . . . . .	28
3.4 PLANAR DOMINATING SET and Annotated Reduction Rules . . . . .	29
3.4.1 The Basic Analysis . . . . .	29
3.4.2 An Improved Analysis . . . . .	31
3.5 FEEDBACK VERTEX SET . . . . .	33
3.6 CLOSEST STRING . . . . .	36
3.7 JUMP NUMBER and Scheduling . . . . .	37
3.8 Exercises . . . . .	43
3.9 Historical Notes . . . . .	46
3.10 Summary . . . . .	47

<b>4</b>	<b>Kernelization</b>	49
4.1	The Basic Method	49
4.2	Heuristics and Applications I	54
4.3	Shrink the Kernel: The Nemhauser–Trotter Theorem	55
4.4	Interleaving	59
4.5	Heuristics and Applications II	60
4.6	Heuristics and Applications III	62
4.7	Definition of a Kernel	63
4.8	$k$ -MAXIMUM LEAF SPANNING TREE	64
4.9	Hereditary Properties and Leizhen Cai’s Theorem	67
4.10	MAXIMUM AGREEMENT FOREST	69
4.10.1	Phylogeny	69
4.10.2	The Algorithm and Correctness	74
4.10.3	Analysis	81
4.11	Exercises	81
4.12	Historical Notes	88
4.13	Summary	89
<b>5</b>	<b>More on Kernelization</b>	91
5.1	Turing Kernelization	91
5.1.1	Rooted $k$ -LEAF OUTBRANCHING	91
5.1.2	$s$ - $t$ and $r$ - $r$ Numberings	92
5.1.3	A Second Combinatorial Bound	95
5.1.4	A Quadratic Kernel	96
5.1.5	$k$ -LEAF OUTBRANCHING and Turing Kernels	99
5.2	Crown Reductions	100
5.3	Exercises	104
5.4	Historical Notes	106
5.5	Summary	106
<b>6</b>	<b>Iterative Compression, and Measure and Conquer, for Minimization Problems</b>	107
6.1	Iterative Compression: The Basic Technique	107
6.2	EDGE BIPARTIZATION	109
6.3	Heuristics and Algorithmic Improvements	112
6.4	Measure and Conquer	113
6.5	FEEDBACK VERTEX SET	117
6.5.1	The Simplified Algorithm	117
6.5.2	The 3-Regular Case	121
6.5.3	An Improved Running Time Using Measure and Conquer	125
6.6	Exercises	128
6.7	Historical Remarks	129
6.8	Summary	130

<b>7</b>	<b>Further Elementary Techniques . . . . .</b>	<b>131</b>
7.1	Methods Based on Induction . . . . .	131
7.1.1	The Extremal Method . . . . .	131
7.1.2	$k$ -LEAF SPANNING TREE, Revisited . . . . .	132
7.2	Greedy Localization . . . . .	134
7.2.1	3-DIMENSIONAL MATCHING . . . . .	135
7.3	Bounded Variable Integer Programming . . . . .	137
7.4	Exercises . . . . .	139
7.5	Historical Notes . . . . .	141
7.6	Summary . . . . .	141
<b>8</b>	<b>Color Coding, Multilinear Detection, and Randomized Divide and Conquer . . . . .</b>	<b>143</b>
8.1	Introduction . . . . .	143
8.1.1	$k$ -PATH . . . . .	144
8.1.2	Dynamic Programming . . . . .	144
8.2	The Randomized Technique . . . . .	144
8.3	De-randomizing . . . . .	145
8.3.1	MULTIDIMENSIONAL MATCHING . . . . .	147
8.4	Speeding Things up with Multilinear Detection . . . . .	148
8.4.1	Group Algebras over $\mathbb{Z}_2^k$ . . . . .	149
8.5	Randomized Divide and Conquer . . . . .	153
8.6	Divide and Color . . . . .	155
8.6.1	The Algorithm and the Kernel . . . . .	156
8.6.2	Probability of a Good Coloring . . . . .	157
8.6.3	Solving a Colored Instance . . . . .	158
8.6.4	Derandomization with Universal Coloring Families . . . . .	160
8.7	Solving MAX- $r$ -SAT ABOVE A TIGHT LOWER BOUND . . . . .	162
8.8	Exercises . . . . .	165
8.9	Historical Notes . . . . .	168
8.10	Summary . . . . .	170
<b>9</b>	<b>Optimization Problems, Approximation Schemes, and Their Relation to FPT . . . . .</b>	<b>171</b>
9.1	Optimization . . . . .	171
9.2	Optimization Problems . . . . .	171
9.3	How FPT and Optimization Problems Relate: Part I . . . . .	172
9.4	The Classes MAX SNP, MIN F <sup>+</sup> $\Pi_1(h)$ and FPT . . . . .	176
9.5	Exercises . . . . .	181
9.6	Historical Notes . . . . .	182
9.7	Summary . . . . .	182

### Part III Techniques Based on Graph Structure

<b>10</b>	<b>Treewidth and Dynamic Programming</b>	185
10.1	Basic Facts About Treewidth	185
10.1.1	Introduction	185
10.1.2	The Basic Definitions	186
10.2	Exercises	189
10.3	Algorithms for Graphs of Bounded Treewidth	190
10.4	Exercises	193
10.5	Bodlaender's Theorem	195
10.6	Exercises	202
10.7	Historical Notes	202
10.8	Summary	203
<b>11</b>	<b>Heuristics for Treewidth</b>	205
11.1	Introduction	205
11.2	Separators	206
11.3	Perfect Elimination	207
11.4	Reduction Rules	209
11.5	Exercises	211
11.6	Historical Notes	211
11.7	Summary	211
<b>12</b>	<b>Methods via Automata and Bounded Treewidth</b>	213
12.1	Introduction	213
12.2	Deterministic Finite Automata	213
12.2.1	Historical Notes	216
12.2.2	Exercises	216
12.3	Nondeterministic Finite Automata	216
12.3.1	Historical Notes	221
12.3.2	Exercises	221
12.4	Regular Languages	222
12.4.1	Historical Notes	226
12.4.2	Exercises	226
12.5	The Myhill–Nerode Theorem and the Method of Test Sets	227
12.5.1	Historical Notes	234
12.5.2	Exercises	235
12.6	Tree Automata	237
12.6.1	Historical Notes	246
12.6.2	Exercises	246
12.7	Parse Trees for Graphs of Bounded Treewidth and an Analogue of the Myhill–Nerode Theorem	247
12.7.1	Historical Notes	263
12.7.2	Exercises	263
12.8	Summary	264

<b>13 Courcelle's Theorem</b> . . . . .	265
13.1 The Basic Theorem . . . . .	265
13.2 Implementing Courcelle's Theorem . . . . .	272
13.3 Seese's Theorem . . . . .	272
13.4 Notes on $MS_1$ Theory . . . . .	274
13.5 Exercises . . . . .	275
13.6 Historical Notes . . . . .	276
13.7 Summary . . . . .	278
<b>14 More on Width-Metrics:</b>	
<b>Applications and Local Treewidth</b> . . . . .	279
14.1 Applications of Width Metrics to Objects Other than Graphs . . . . .	279
14.1.1 Logic . . . . .	280
14.1.2 Matroid Treewidth and Branchwidth . . . . .	281
14.1.3 Knots, Links, and Polynomials . . . . .	282
14.2 Local Treewidth . . . . .	284
14.2.1 Definitions . . . . .	284
14.2.2 The Frick–Grohe Theorem . . . . .	286
14.3 Exercises . . . . .	288
14.4 Historical Notes . . . . .	288
14.5 Summary . . . . .	289
<b>15 Depth-First Search and the Plehn–Voigt Theorem</b> . . . . .	291
15.1 Depth-First Search . . . . .	291
15.2 Exercises . . . . .	293
15.3 Bounded-Width Subgraphs, the Plehn–Voigt Theorem, and Induced Subgraphs . . . . .	295
15.4 Exercises . . . . .	299
15.5 Historical Remarks . . . . .	300
15.6 Summary . . . . .	300
<b>16 Other Width Metrics</b> . . . . .	301
16.1 Branchwidth . . . . .	301
16.2 Basic Properties of Branchwidth . . . . .	302
16.3 Dynamic Programming and Branchwidth . . . . .	303
16.4 Fast Matrix Multiplication and Dynamic Programming . . . . .	306
16.4.1 Improving Things Using Matrix Multiplication . . . . .	307
16.5 Cliquewidth and Rankwidth . . . . .	309
16.6 Directed Graphs . . . . .	312
16.7 $d$ -Degeneracy . . . . .	313
16.8 Exercises . . . . .	315
16.9 Historical Notes . . . . .	316
16.10 Summary . . . . .	316

**Part IV Exotic Meta-techniques**

<b>17</b>	<b>Well-Quasi-Orderings and the Robertson–Seymour Theorems . . .</b>	<b>319</b>
17.1	Basics of Well-Quasi-Orderings Theory . . . . .	319
17.2	Connections with Automata Theory and Boundaried Graphs . .	333
17.3	Exercises . . . . .	336
17.4	Historical Notes . . . . .	337
17.5	Summary . . . . .	338
<b>18</b>	<b>The Graph Minor Theorem . . . . .</b>	<b>339</b>
18.1	Introduction . . . . .	339
18.2	Excluding a Forest . . . . .	340
18.3	Thomas’ Lemma . . . . .	342
18.4	The Graph Minor Theorem . . . . .	345
18.5	The Immersion and Topological Orders . . . . .	347
18.6	The Summary . . . . .	348
18.7	Exercises . . . . .	349
18.8	Historical Notes . . . . .	350
18.9	Summary . . . . .	350
<b>19</b>	<b>Applications of the Obstruction Principle and WQOs . . . . .</b>	<b>351</b>
19.1	Methods for Tractability . . . . .	351
19.2	Effectivizations of Obstruction-Based Methods . . . . .	357
	19.2.1 Effectivization by Self-reduction . . . . .	357
	19.2.2 Effectivization by Obstruction Set Computation . . .	361
19.3	The Irrelevant Vertex Technique . . . . .	364
19.4	Protrusions . . . . .	365
	19.4.1 Informal Description . . . . .	366
	19.4.2 Overview of the Protrusion Based Meta-kernelization Results . . . . .	367
	19.4.3 Overview of the Methods . . . . .	368
19.5	Exercises . . . . .	369
19.6	Historical Remarks . . . . .	371
19.7	Summary . . . . .	372

**Part V Hardness Theory**

<b>20</b>	<b>Reductions . . . . .</b>	<b>375</b>
20.1	Introduction . . . . .	375
20.2	Parameterized Reductions . . . . .	377
20.3	Exercises . . . . .	380
20.4	Historical Notes . . . . .	382
20.5	Summary . . . . .	382

<b>21</b>	<b>The Basic Class <math>W[1]</math> and an Analog of Cook's Theorem</b> . . . . .	383
21.1	Introduction . . . . .	383
21.2	Short Turing Machine Acceptance . . . . .	384
21.3	Exercises . . . . .	401
21.4	Historical Notes . . . . .	405
21.5	Summary . . . . .	406
<b>22</b>	<b>Some Other <math>W[1]</math> Hardness Results</b> . . . . .	407
22.1	PERFECT CODE and the Turing Way to $W$ -Membership . . . . .	407
22.2	The VC Dimension . . . . .	412
22.3	Logic Problems . . . . .	414
22.4	Exercises . . . . .	424
22.5	Historical Notes . . . . .	426
22.6	Summary . . . . .	426
<b>23</b>	<b>The <math>W</math>-Hierarchy</b> . . . . .	427
23.1	Introduction . . . . .	427
23.2	The Normalization Theorem . . . . .	429
23.2.1	The DOMINATING SET Reduction . . . . .	429
23.2.2	The Proof of the Normalization Theorem . . . . .	434
23.3	Monotone and Antimonotone Formulas . . . . .	444
23.4	The $W^*$ -Hierarchy and Generalized Normalization to $W^*[t]$ . . . . .	446
23.4.1	$W^*$ Normalization . . . . .	447
23.4.2	An Improved Characterization of $W[1]$ . . . . .	448
23.5	Exercises . . . . .	455
23.6	Historical Notes . . . . .	459
23.7	Summary . . . . .	459
<b>24</b>	<b>The Monotone and Antimonotone Collapse Theorems: MONOTONE <math>W[2t + 1] = W[2t]</math> and ANTIMONOTONE <math>W[2t + 2] = W[2t + 1]</math></b> . . . . .	461
24.1	Results . . . . .	461
24.2	Historical Notes . . . . .	471
24.3	Summary . . . . .	471
<b>25</b>	<b>Beyond <math>W[t]</math>-Hardness</b> . . . . .	473
25.1	$W[P]$ and $W[SAT]$ . . . . .	473
25.2	Exercises . . . . .	485
25.3	Historical Remarks . . . . .	488
25.4	Summary . . . . .	489

<b>26</b>	<b>Fixed Parameter Analogues of PSPACE and <math>k</math>-Move Games</b>	491
26.1	Introduction	491
26.2	$k$ -Move Games	491
26.3	AW[*]	495
26.4	Relational Databases	499
26.5	Exercises	502
26.6	Historical Notes	506
26.7	Summary	507
<b>27</b>	<b>Provable Intractability: The Class XP</b>	509
27.1	Introduction and X Classes	509
27.2	Pebble Games	510
27.3	Other XP-Completeness Results	517
27.4	Exercises	518
27.5	Historical Remarks	518
27.6	Summary	519
<b>28</b>	<b>Another Basis for the <math>W</math>-Hierarchy and the Tradeoff Theorem</b>	521
28.1	Results	521
28.2	Exercises	530
28.3	Historical Notes	531
28.4	Summary	531
<b>Part VI Approximations, Connections, Lower Bounds</b>		
<b>29</b>	<b>The <math>M</math>-Hierarchy, and XP-Optimality</b>	535
29.1	Introduction	535
29.2	The $W$ -Hierarchy and SUBEXP Time	535
29.3	The $M$ -Hierarchy	541
29.4	Parametric Miniatures	542
29.5	The Sparsification Lemma	546
29.5.1	Statement of the Lemma	546
29.5.2	Using the Lemma for $O^*(2^{o(k)})$ Lower Bounds	546
29.5.3	No $O^*((1 + \epsilon)^k)$ Algorithm for VERTEX COVER	552
29.6	The Proof of the Sparsification Lemma	553
29.7	XP-Optimality	556
29.8	Treewidth Lower Bounds	557
29.9	The Strong Exponential Time Hypothesis	559
29.10	Subexponential Algorithms and Bidimensionality	561
29.11	Exercises	565
29.12	Historical Notes	567
29.12.1	Prehistory	567
29.12.2	History	569
29.13	Summary	570

<b>30</b>	<b>Kernelization Lower Bounds</b>	571
30.1	Introduction	571
30.2	A Generic Engine for Lower Bounds of Kernels	572
30.3	Applications of OR-Composition	577
30.3.1	Notes on Turing Kernels	579
30.4	AND Composition	579
30.5	Proof of Drucker's Theorem	582
30.5.1	Statistical Distance and Distinguishability	582
30.5.2	Proof of Theorem 30.5.2	583
30.6	Cross-composition	588
30.7	Unique Identifiers	592
30.8	Sharpening Bounds	595
30.9	The Packing Lemma	596
30.10	Co-nondeterminism	597
30.10.1	Co-nondeterministic Composition, and $k$ -RAMSEY	597
30.10.2	Co-nondeterministic Cross-composition and Hereditary Properties	602
30.11	Weak Composition and Sharp Lower Bounds Revisited	606
30.12	Miniature Parametric Miniatures	612
30.13	Exercises	615
30.14	Historical Notes	619
30.15	Summary	619

## Part VII Further Topics

<b>31</b>	<b>Parameterized Approximation</b>	623
31.1	Introduction	623
31.2	Pathwidth	628
31.3	Treewidth	633
31.4	Parameterized Online Algorithms and Incremental Computing	635
31.4.1	Coloring	636
31.4.2	Online Coloring Graphs of Bounded Pathwidth	636
31.5	Parameterized Inapproximability	639
31.6	Exercises	641
31.7	Historical Notes	643
31.8	Summary	644

<b>32</b>	<b>Parameterized Counting and Randomization</b>	645
32.1	Introduction	645
32.2	Classical Counting Problems and #P	646
32.3	#W[1]—A Parameterized Counting Class	648
32.4	The Counting Analog of Cook's Theorem	649
32.5	The #W-Hierarchy	657
32.6	An Analog of Valiant's Theorem	658
32.7	Parameterized Randomization	663
32.8	Exercises	670
32.9	Historical Notes	671
32.10	Summary	673
<b>Part VIII Research Horizons</b>		
<b>33</b>	<b>Research Horizons</b>	677
33.1	The Most Infamous	677
33.2	Think Positive!	679
33.3	More Tough Customers	681
33.4	Exemplars of Programs	684
<b>Part IX Appendices</b>		
<b>34</b>	<b>Appendix 1: Network Flows and Matchings</b>	689
34.1	Network Flows	689
34.1.1	Basic Results	689
34.1.2	Exercises	691
34.1.3	The Ford–Fulkerson Algorithm	692
34.1.4	Matching Theory	693
34.2	Berge's Criterion	695
34.3	Edmonds' Algorithm	697
34.4	Bipartite Graphs and VERTEX COVER	699
34.5	Matching and Co-graphic Matroids	700
34.6	The MATROID PARITY PROBLEM	704
<b>35</b>	<b>Appendix 2: Menger's Theorems</b>	705
35.1	Connectivity	705
35.2	Exercises	707
<b>References</b>		709
<b>Index</b>		745